US Patent 5,611,049, Claim 1:

- In a network of digital computers that includes a first plurality of Network Distributed Cache ("NDC") sites, each NDC site including an NDC that has an NDC buffer, a method for projecting images of a stored dataset from an NDC server terminator site into a second plurality of NDC client terminator sites in response to requests to plurality of client sites respectively to the second plurality of NDC concurrently access such stored dataset transmitted from a third client terminator sites, the method comprising the steps of:
- (a) the NDC receiving the request to access data in the stored dataset;
- if a projected image of data requested from the stored dataset is already (b) the NDC checking the NDC buffer at this NDC site to determine present there;
- data from this NDC site downstream to another NDC site closer to the image of all data requested from the stored dataset, and if the NDC site (c) if the NDC buffer of this NDC site does not contain a projected stored dataset, the NDC of this NDC site transmitting a request for receiving the request is not the NDC server terminator site for the NDC server terminator site for the stored dataset than the present NDC site;
- stored dataset, the NDC of the NDC server terminator site accessing the stored dataset to project an image of the requested data into the NDC image of all data requested from the stored dataset, and if the NDC site (d) if the NDC buffer of this NDC site does not contain a projected receiving the request is the NDC server terminator site for the buffer of the NDC server terminator site;
- downstream NDC site receiving the request contains a projected image (e) repeating the steps (a) through (d) until the NDC buffer of the of all requested data;
- data arrives at the NDC client terminator site, each NDC site that retums data upstream to the requesting NDC site retaining a copy of the returned data that the returning NDC site may subsequently transmit to an NDC returned the data, whereby images of the stored dataset may be projected (f) each successive NDC site, having obtained a projected mings of the requested data upstream to the NDC the requested data upstream to the NDC concurrently from a single NDC site into the second plurality of NDC site from which the NDC site received the request until the requested site other than the NDC site to which the returning NDC site first client terminator sites; and
- (g) the NDC client terminator site, upon receiving the requested data, returning the requested data to the client site that requested access to the stored dataset.

BordotManagus - Thine Ways to Deliver Caphad Purturmanse to Your Istrand and Viornial Users (AppNose) N.W. 50 BE PRODUCTING PRODUCT RECONSES COMPITITIVE INFO MELATED PRODUCTS EVAL FW SUFFORT SMORTHOMS SITE MAD HORFIS HOW TO BUY SEPTEMBER1997 11340# N DV 😅 📗 WHAT'S NEW

Three Ways to Deliver Cached Intranet and Internet Users Performance to Your

Senior Research Engineer Advanced Development Group RON LEE

enhance network performance. Comprehensive security restrictions, access econols, and content filtering performance penalty in an environment where users are already frustrated by busy. Web servers and lang their systems using the mass cost-effective means available. Yet the wickspread deployment of Internet and intrince connections has imposed new requirements that seem to be in confluct with these efforts to Network engineers and administrators are constantly trying to squeeze the highest performance out of are crucial aspects of securing the intranet and connecting to the Internet, but they exact an additional respiranse times.

infrastructure and offset the performance penalty you pay for the necessary security exarteds and filtering. Novell's Border-Maragox includes an Internet object cache that significatedly increases the speed of web access. In the process, this technology provides a performance foundation to support your network

This AppNote provides an overview of BonderManager's caching technology and discusses the advantages of eaching in Intranet and Internet environments. It then describes three applications of Nevell's Interest object cache that provide significant benefits to intraset and Internet users:

---- Proxy carling

..... Proxy cardie hierarchies

asa Web server acceleration

For more information on BonderManager and other AppNotes regarding these technicisgiess, visit the Novell World Wide site at http://www.novell.com/burde/manages

What is Caching?

During the 1960s, competer designers discovered that much of the program code their systems were executing was extremely repetitive--mail portions of the code would be processed over and over again. Using this insight to their advantage, they began storing the repetitive portions of their programs in a

happidenny rowel combordermanagements from the 11st Houtertoon 4.65.26 PM;

NCT010657

(a) the NDC receiving the request to access data in the stored dataset;

(b) the NDC checking the NDC buffer at this NDC site to determine if a projected image of data requested from the stored dataset is already present there; (c) if the NDC buffer of this NDC site does not contain a projected image of all data requested from the stored dataset, and if the NDC site receiving the request is not the NDC server terminator site for the stored dataset, the NDC of this NDC site transmitting a request for data from this NDC site downstream to another NDC site closer to the NDC server terminator site for the stored dataset than the present NDC site;

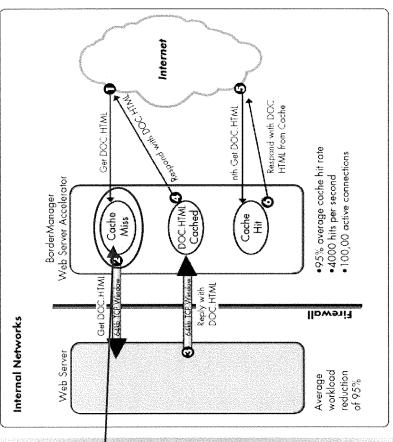
(d) if the NDC buffer of this NDC site does not contain a projected image of all data requested from the stored dataset, and if the NDC site receiving the request is the NDC server terminator site for the stored dataset, the NDC of the NDC server terminator site accessing the stored dataset to project an image of the requested data into the NDC buffer of the NDC server terminator site;

Web Server Acceleration

Web servers can be a bottleweck in your intranet or internet infrastructures. Typical web servers quickly run out of connection copacity and tend to produce slow response times. In sites where performance is important, the only options usually considered are to upgrade to a more expensive web server system or to split the content set across multiple web servers. Neither of these options make sence when caching offers such an elegant, cost-effective means to overcome the problem.

Configured as a web server accelerator, Novell's futuret object cache eliminates the web server bottleneck by placing a dedicated cache in front of the web server and handling requests for all of the web server's cacheable content directly from its own cache. Caching is the obvious solution because typical web sizes are constructed with approximately 95-100 percent cacheable content. Once this material is fatched from the web server and cacheable in the web server accelerator, the accelerator can handle all of the requests for that content. This leaves the small percentage of dynamic requests to be based through "the accelerator for the origin web server to process (see Figure 8).

Physics B. The web server accelerator offloads over 90 percent of the web server's workload and responds to requests at cached speeds



- A browser issues a request for a file named DOC HTML. This request is received by the web server accelerator. In this case, the request
 results in a "cache miss" because the web server accelerator has never serviced a request for that document before.
 - The web server accelerator initiates a request for DOC.HTML from your web server on behalf of the browser.
- 2. He origin web server responds to the web server accelerant's request by sending DOCHFUL. This transmission is much faster than a response to a thorster due to the web server accelerant's request by sending DOCHFUL. This transmission is much faster than a response to a thorster due to the web server accelerant's optimizeneity writing that can receive up to 64KB at one time and that stays open to receive multiple responses. The web server accelerant their places DOCHFUL in its cache.
- . The web server accelerator responds to the original browser request with DOC.HTML.
- 5. Now when the same browser (or any other browser) is sue a request for DOC.HTML, the request results in a "tacke hill" because the web server accelerator has kept a copy of the document in its cache.
 - In this case, the web server accelerator replies immediately to the browner request because it has DOC.HTML in cache. The proxy's response eliminates the need to foth the document again from the origin web server.

(a) the NDC receiving the request to access data in the stored dataset;

if a projected image of data requested from the stored dataset is already (b) the NDC checking the NDC buffer at this NDC site to determine present there;

data from this NDC site downstream to another NDC site closer to the image of all data requested from the stored dataset, and if the NDC site (c) if the NDC buffer of this NDC site does not contain a projected stored dataset, the NDC of this NDC site transmitting a request for receiving the request is not the NDC server terminator site for the NDC server terminator site for the stored dataset than the present NDC site;

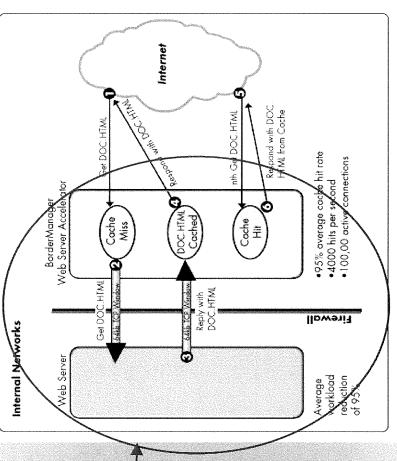
stored dataset, the NDC of the NDC server terminator site accessing the image of all data requested from the stored dataset, and if the NDC site stored dataset to project an image of the requested data into the NDC (d) if the NDC buffer of this NDC site does not contain a projected receiving the request is the NDC server terminator site for the buffer of the NDC server terminator site;

Web Server Acceleration

Configured as a web server accelerator, Novell's bremet object cache eliminates the web server bottleneck by placing a dedicated cache in front of the produce slow response times. In sites where performance is important, the only options usually considered are to upgrade to a more expensive web Web servers can be a bottleneck in your intranet or humest infrastructures. Typical web servers quickly run out of connection capacity and tend to server system or to split the content set across multiple web servers. Neither of these options make sense when caching offers such an elegant, cost-effective means to overcome the problem

in the web server accelerator, the accelerator can handle all of the requests for that content. This leaves the small percentage of dynamic requests to be typical web sizes are constructed with approximately 95-100 percent cacheable content. Once this material is forthed from the web server and cached web server and handling requests for all of the web server's cacheable content directly from its own cache. Caching is the obvious solution because passed through "the accelerator for the oxigin web server to process (see Figure 8).

Pigure 8. The web server accelerator affloads over 90 percent of the web server's workload and responds to requests at carbed speeds



- 1. A browser isques a request for a file named DOC.HTML. This request is received by the web server accelerator. In this case, the request results in a "tache miss" because the web server accelerator has never serviced a request for that document before
 - The web server accelerator initiates a request for DOC.HIML from your web server on behalf of the browser.
- response to a browser due to the web server accelerator's optimized receive window that can receive up to 64KB at one time and that stays The origin web server responds to the web server accelerator's request by sending DOC.HTML. This transmission is ranch faster than a open to receive multiple responses. The web server accelerator then places DOC.HTML in its cache
 - The web server accelerator responds to the original browser request with DOC HTML
- Now when the same browser (or any other browser) issues a request for DOC HIML, the request results in a "cache him" because the web server accelerator has kept a copy of the document in its cache.
 - In this case, the web server accelerator replies from ediately to the browser request because it has DOC.HTML in cache. The proxy's response eliminates the need to fetch the document again from the origin web server

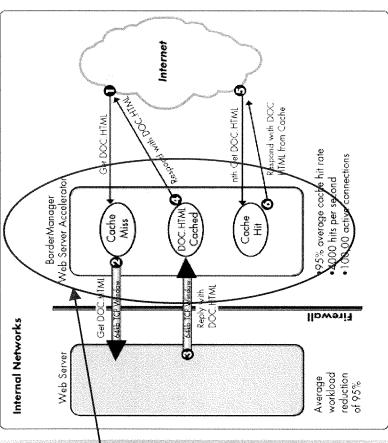
- (a) the NDC receiving the request to access data in the stored dataset;
- if a projected image of data requested from the stored dataset is already (b) the NDC checking the NDC buffer at this NDC site to determine present there;
- data from this NDC site downstream to another NDC site closer to the image of all data requested from the stored dataset, and if the NDC site (c) if the NDC buffer of this NDC site does not contain a projected stored dataset, the NDC of this NDC site transmitting a request for receiving the request is not the NDC server terminator site for the NDC server terminator site for the stored dataset than the present NDC site;
- stored dataset, the NDC of the NDC server terminator site accessing the image of all data requested from the stored dataset, and if the NDC site stored dataset to project an image of the requested data into the NDC (d) if the NDC buffer of this NDC site does not contain a projected receiving the request is the NDC server terminator site for the buffer of the NDC server terminator site;

Web Server Acceleration

produce slow response times. In sizes where performance is important, the only options usually considered are to upgrade to a more expensive web Web servers can be a bottleneck in your intremet or Internet infrastructures. Typical web servers quickly run out of connection capacity and tend to server system or to split the content set across multiple web servers. Neither of these options make sense when caching offers such an elegant, cost-effective means to overcome the problem

n the web server accelerator, the accelerator can handle all of the requests for that content. This leaves the small percentage of dynamic requests to be Configured as a 1986 server accelerator, Movell's Internet object cache eliminates the 1986 server bortleweck by placing a dedicated cache in front of the web server and handling requests for all of the web server's cacheable content directly from its own cache. Caching is the obvious solution because typical web sites are constructed with approximately 95-100 percent cacheable content. Once this material is fatched from the web server and cached passed through" the accelerator for the origin web server to process (see Figure 8).

Figure 8: The web server accelerator officeasi over 90 percent of the web server's workload and responsis to requests at carbed speeds



- A browser issues a request for a file named DOC HTML. This request is received by the web server accelerator. In this case, the request results in a "cache miss" because the web server accelerator has never serviced a request for that document before
 - The web server accelerator initiates a request for DOC HTML from your web server on behalf of the browser.
- response to a browser due to the web server accelerator's optimized receive window that can receive up to 64KB at one time and that stays The origin web server responds to the web server accelerator's request by sending DOC.HIML. This transmission is much faster than a open to receive multiple responses. The web server accelerator then places DOC HTML in its cache.
 - The web server accelerator responds to the original browser request with DOC.HTML
- Now when the same browser (or any other browser) issues a request for DOC.HIML, the request results in a "tache him" because the web server accelerator has kept a copy of the document in its cache
 - in this case, the web server accelerator replies immediately to the browser request because it has DOC.HTML in cache. The proxy's response eliminates the need to fetch the document again from the origin web server. ö

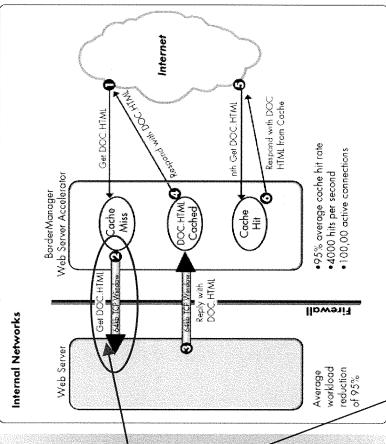
- (a) the NDC receiving the request to access data in the stored dataset;
- (b) the NDC **checking the NDC buffer** at this NDC site to determine if a **projected image of data requested** from the stored dataset is already present there;
- (c) if the NDC buffer of this NDC site does not contain a projected image of all data requested from the stored dataset, and if the NDC site receiving the request is not the NDC server terminator site for the stored dataset, the NDC of this NDC site transmitting a request for data from this NDC site downstream to another NDC site closer to the NDC server terminator site for the stored dataset than the present NDC site.
- (d) if the NDC buffer of this NDC site does not contain a projected image of all data requested from the stored dataset, and if the NDC site receiving the request is the NDC server terminator site for the stored dataset, the NDC of the NDC server terminator site accessing the stored dataset to project an image of the requested data into the NDC buffer of the NDC server terminator site.

Web Server Acceleration

Web servers can be a bottleneck in your intranst of interactuations. Typical web servers quickly run out of connection capacity and tend to produce slow response times, in sites where performent, the only options usually considered are to upgrade to a more expensive web server system or to split the content set across multiple web servers. Neither of these options make sence when caching offers such an elegant, cost-effective means to overcome the problem.

Configured as a web server accelerator, Novell's brannet object cache eliminates the web server bothleneck by placing a dedicated cache in front of the web server and handling requests for all of the web server's cacheable content directly from its own cache. Caching is the obvious solution because typical web sites are constructed with approximately 95-100 percent cacheable content. Once this material is fetched from the web server and cacheable in the web server accelerator, the accelerator can handle all of the requests for that content. This leaves the small percentage of dynamic requests to be bassed through' the accelerator for the origin web server to process (see Figure 8).

Typure B. The web server accelerator offloads over 90 percent of the web server's workload and responds to requests at cached speeds



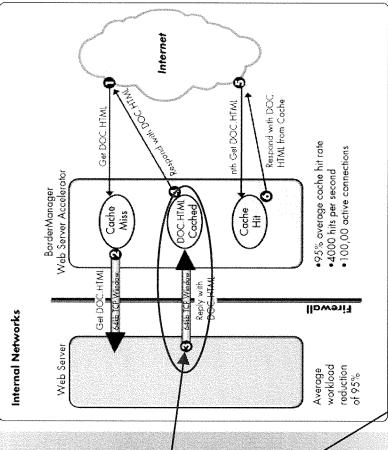
- 1) A browser issues a request for a file named DOC HTIML. This request is received by the web server accelerator. In this case, the request suits in a "cache miss" because the web server accelerator has never serviced a request for that document before.
 - 2. The web server accelerator initiates a request for DOC HTML from your web server on behalf of the browser.
 3. The origin web server responds to the web server accelerators request by sending DUC HTML. THIS transmission is much faster than a response to a towner due to the web server accelerator's optimized reserve window that can receive up to 64KB at one time and that stays open to receive multiple responses. The web server accelerator them places DOC HTML in acceler.
 - The web server accelerator responds to the original browser request with DOC.HTML.
- Now when the same browser (or any other browser) is sue, a request for DOC.HTML, the request results in a "value hit" because the web server accelerator has kept a copy of the document in its eache.
- In this case, the web server accelerator replies immediately to the browser request because it has DOC.HTML in cache. The proxy's response eliminates the need to fetch the document again from the origin web server.

- (a) the NDC receiving the request to access data in the stored dataset;
- if a projected image of data requested from the stored dataset is already (b) the NDC checking the NDC buffer at this NDC site to determine present there;
- data from this NDC site downstream to another NDC site closer to the image of all data requested from the stored dataset, and if the NDC site (c) if the NDC buffer of this NDC site does not contain a projected stored dataset, the NDC of this NDC site transmitting a request for receiving the request is not the NDC server terminator site for the NDC server terminator site for the stored dataset than the present NDC site;
- stored dataset, the NDC of the NDC server terminator site accessing the stored dataset to project an image of the requested data into the NDC image of all data requested from the stored dataset, and if the NDC site (d) if the NDC buffer of this NDC site does not contain a projected receiving the request is the NDC server terminator site for the buffer of the NDC server terminator site;

Web Server Acceleration

produce slow response times. In sizes where performance is important, the only options usually considered are to upgrade to a more expensive web Web servers can be a bottleweck in your intravet or internet infractuotines. Typical web servers quickly run out of connection capacity, and tend to server system or to split the content set across multiple web servers. Neither of these options make sense when caching offers such an elegant, cost-effective means to overcome the problem Configured as a web server accelerator, Novell's Internet object cache eliminates the web server bottleneck by placing a dedicated cache in front of the web server and handling requests for all of the web server's cacheable content directly from its own cache. Caching is the obvious solution because typical web size are constructed with approximately 95-100 percent cacheable content. Once this material is fathed from the web server and cached in the web server screlerator, the accelerator can handle all of the requests for that contact. This leaves the small purcertage of dynamic requests to be passed through "the accelerator for the origin web server to process (see Figure 8).

Pigure B. The web server accelerator affloads over 90 percent of the web server's workload and responds to requests at carbed speeds



A browser issues a request for a file named DOC, HTML. This request is received by the web server accelerator. In this case, the request ssults in a "tache miss" because the web server accelerator has never serviced a request for that document before

he web server accelerator initiates a request for DOC.HTML from your web server on behalf of the browser

- response to a browser due to the web server accelerator's optimized receive window that can receive up to 64KB at one time and that stays The origin web server responds to the web server accelerator's request by sending DOC.HTML. This transmission is much faster than a open to receive multiple responses. The web server accelerator than places DOC.HIML in its cache
- the web server accelerator responds to the original browser request with DUC.HIM
- 5. Now when the same browser (or any other browser) issues a request for DOC.HIML, the request results in a "tache hit" because the web server accelerator has kept a copy of the document in its cache
- in this case, the web server accelerator replies immediately to the browser request because it has DOC.HTML in cache. The proxy's response eliminates the need to fetch the document again from the origin web server.